Available online at www.sciencedirect.com

ScienceDirect

Journal of Electrical Systems and Information Technology 1 (2014) 212–222

www.elsevier.com/locate/jesit

LFC based adaptive PID controller using ANN and ANFIS techniques

Mohamed I. Mosaad [a,*], Fawzan Salem [b]

[a] *Higher Technological Institute (HTI), Egypt*
[b] *Electronics Research Institute (ERI), Egypt*

Available online 12 December 2014

Abstract

This paper presents an adaptive PID Load Frequency Control (LFC) for power systems using Neuro-Fuzzy Inference Systems (ANFIS) and Artificial Neural Networks (ANN) oriented by Genetic Algorithm (GA). PID controller parameters are tuned off-line by using GA to minimize integral error square over a wide-range of load variations. The values of PID controller parameters obtained from GA are used to train both ANFIS and ANN. Therefore, the two proposed techniques could, online, tune the PID controller parameters for optimal response at any other load point within the operating range. Testing of the developed techniques shows that the adaptive PID-LFC could preserve optimal performance over the whole loading range. Results signify superiority of ANFIS over ANN in terms of performance measures.

Keywords: LFC; ANFIS; ANN; GA; Adaptive PID control

1. Introduction

Frequency control is a key stability criterion in power systems. To ensure stability of the power system, active power balance and constant frequency are required. Frequency depends on active power balance. If any change occurs in active power in power systems, frequency cannot be hold in its rated value and oscillations are increased in both power and frequency. Thus, the system is subjected to a serious instability problem. With proper design of Load Frequency Control (LFC) systems, stability of power system networks is improved.

Several controllers have been introduced for power system LFC to achieve a better dynamic performance. Examples for these controllers are proportional integral (PI) control (Nanda and Kaul, 1978), state feedback control (Elgerd and Fosha, 1970), and output feedback control (Aldden and Trinh, 1994). Recently, intelligent control techniques such as fuzzy control are applied to LFC problem (Hsu and Cheng, 1991). Most of the above controllers depend on fixed gain control parameters, like PI controller.

* Corresponding author.
E-mail addresses: m_i_mosaad@hotmail.com (M.I. Mosaad), fawzan@eri.sci.eg (F. Salem).
Peer review under the responsibility of Electronics Research Institute (ERI).

Production and hosting by Elsevier

http://dx.doi.org/10.1016/j.jesit.2014.12.004

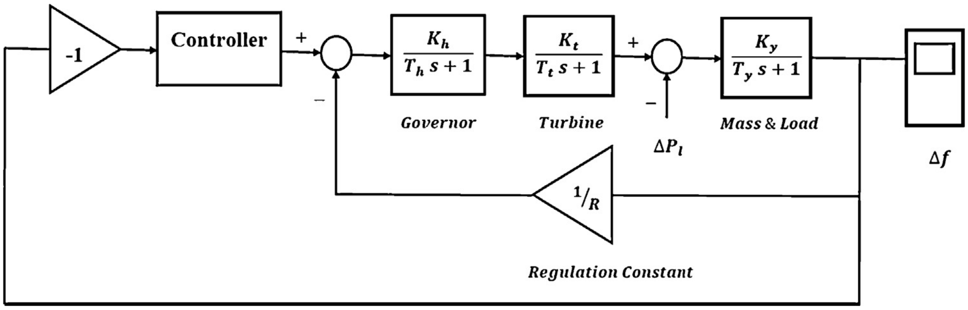

Fig. 1. One area power generating model.

However, fixed gain controllers are designed at nominal operating points and may not be suitable for all operating conditions. Therefore, adaptive gain scheduling approaches have been proposed for LFC (Talaq and Al-Basri, 1999), where an adaptive fuzzy gain scheduling scheme for conventional PI and optimal controllers has been proposed and tested for off-nominal operating conditions.

Since it is difficult to examine all the input–output data from a system to search for a number of proper rules for the fuzzy system, automatic design of the fuzzy rules in the fuzzy gain scheduling control approach by Genetic Algorithm (GA) is proposed in (Juang and Lu, 2002). In this approach, a fuzzy system is used to adaptively decide the integral controller gain according to the area control errors (ACE) and their changes.

Moreover, an Artificial Neural Networks (ANN) approach is also applied for LFC (Sabahi et al., 2007). In this method, a modified dynamic neural network (MDNN) controller was designed to LFC application in power system for generating electricity with good quality. MDNN also was used to identify the model simultaneously with control process.

On the other hand, Jang (1993) coined the term ANFIS which stands for Adaptive Network-based Fuzzy Inference Systems, or semantically equivalently, Adaptive Neuro-Fuzzy Inference Systems. ANFIS is an intelligent regime comprising an adaptive network performing the function of a Sugeno-type fuzzy model (Jang et al., 1997; Cox, 1992). Optimization algorithms employed through adaptive networks make the system performance similar to a targeted training data set. ANFIS combines the optimization strength of adaptive networks with the ability of fuzzy systems to handle vague situations and process uncertain data. Such attribute enabled ANFIS to find many immediate engineering applications including, but not limited to, decision making, problem solving, pattern recognition, nonlinear mapping, system modeling, and adaptive control.

In this paper, both ANN and ANFIS are employed to present a new adaptive PID Load Frequency Control for power systems. The parameters of the PID controller are tuned off-line by using Genetic Algorithm technique in order to minimize the integral of square of error over a wide-range of operating conditions. Several load changes covering the range from −0.5 p.u. to +0.5 p.u. are considered for training. The data obtained from Genetic Algorithm are used to train both ANN and ANFIS that could produce PID controller parameters for optimal response at any other load value within the operating range. Testing the developed techniques show that the adaptive PID Load Frequency Control could preserve optimal performance over the whole loading range with superiority to ANFIS in terms of several performance measures such as Integral of Absolute of Errors (IAE), Integral of Time multiplied Absolute of Errors (ITAE), and Integral of Square of Errors (ISE).

2. Problem formulation

The aim of LFC is to maintain a real power balance in the power system through controlling the system frequency. Whenever the real power demand changes, a frequency change occurs. This frequency error is amplified, mixed and changed to a command signal which is sent to turbine governor. The governor works to restore the balance between the input and output by changing the turbine output. This method is also referred as Megawatt frequency or Power–frequency (P–f) Control (Saadat, 2005). The block diagram for single area non-reheat type is depicted in Fig. 1.

Basically, electric power system components are non-linear; therefore a linearization around a nominal operating point is usually performed to get a linear system model which is used in the controller design process. The power generating units include turbines, generators, rotating mass and load, hydraulic valve actuator and governors.

2.1. Turbines

A turbine unit in power systems is used to transform the natural energy, such as the energy from steam or water, into mechanical power (ΔP_T) that is supplied to the generator, in general. The non-reheat turbine can be represented as (Kundur, 1994):

$$G_{NR}(s) = \frac{\Delta P_T(s)}{\Delta P_V(s)} = \frac{1}{T_t s + 1} \quad (1)$$

where T_t is the time delay that occurs between switching the valve and producing the turbine torque and ΔP_V, is the valve/gate position change.

2.2. Generators

A generator unit in power systems converts the mechanical power received from the turbine into electrical power, but for LFC, a focus will be on the rotor speed output (frequency of the power system) of the generator instead of the energy transformation. In normal steady state, the turbine mechanical power P_T keeps balance with the electromechanical air-gap power P_G resulting in zero acceleration and a constant speed or frequency. Any changes in these powers will upset the above balance. The generator power increment ΔP_G depends on the changes in the load, ΔP_L, fed from the generator. The generator always adjusts its output to meet the demand changes ΔP_G.

2.3. Governors

Governors are the units that are used in power systems to sense the frequency bias caused by the load change and eliminate it by varying the inputs of the turbines. The real power, in a power system is controlled by controlling the driving torques of the individual turbines of the system. For the single area non-reheat thermal system considered in this study, the conventional PID controller is replaced with an adaptive PID controller whose controller parameters are tuned on-line according to load change ΔP_L using ANN and ANFIS.

3. Genetic algorithms

Genetic algorithms (GAs) are biologically inspired techniques used for optimization. These algorithms encode a potential solution to a specific problem on a simple chromosome like data structure and apply recombination operators to these structures so as to preserve critical information (Haupt and Haupt, 1998).

The GA begins, like any other optimization algorithm, by defining the optimization variables, the fitness function, and the fitness. It ends like other optimization algorithms too, by testing for convergence.

The general steps implemented when using GAs are:

1. Generate a random initial population.
2. Create the new population by applying the selection and reproduction operators to select pairs of strings. The number of pairs is the population size divided by two, so the population size will remain constant between generations.
3. Apply the crossover operator to the pairs of the strings of the new population.
4. Apply the mutation operator to each string in the new population.
5. Replace the old population with the newly created population.
6. Copy the best-fitted individuals to the newly created population to warrantee evolution.
7. If the number of iterations is less than the maximum go to step two, else stop OR If the fitness of the best result does not get better over certain number of iteration, then stop.

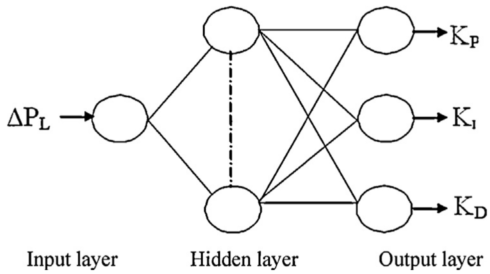

Fig. 2. Schematic diagram of the construction of the adopted neural network model.

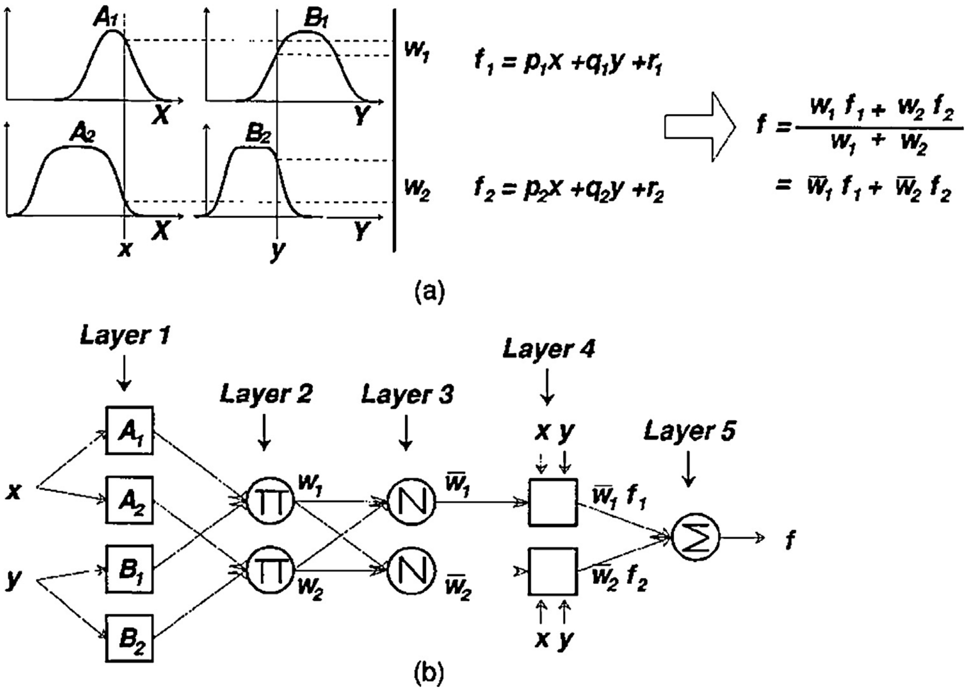

Fig. 3. (a) Two-input, one-output, two-rule Sugeno model and (b) equivalent ANFIS structure.

3.1. Implementation of GAs in the optimization problem

To solve the optimization problem using GAs, all of the possible solutions have to be coded in chromosomes, that is the three control parameters (K_P, K_I and K_D). Series binary coding is used in this paper. Next a fitness function (ISE) is calculated, to compare the chromosomes which have to be defined. Therefore, to calculate the fitness of a chromosome, the optimization function has to be calculated using the information in the chromosome.

4. Adaptive control techniques

Adaptive control is the attempt to redesign the controller according to its performance and to tune its parameters automatically. Both ANN and ANFIS can be utilized to fine tune the PID controller parameters online to be an adaptive PID controller. In the following an overview is given for both of the two techniques.

4.1. Artificial Neural Network (ANN)

Artificial Neural Networks are considered as a relatively new information processing technique. They can be defined as “a computing system made up of a number of simple, highly interconnected processing elements, which

Table 1
Control parameters from GAs, ANN and ANFIS.

ΔP_L	Algorithm								
	GA			ANN			ANFIS		
	K_P	K_I	K_D	K_P	K_I	K_D	K_P	K_I	K_D
−0.5	0.834	1.523	0.215	0.797	1.555	0.2	0.8342	1.5228	0.2148
−0.45	1.194	2.211	0.262	1.234	2.192	0.269	1.1944	2.2114	0.2619
−0.4	2.236	3.754	0.405	2.238	3.657	0.428	2.2355	3.7541	0.4054
−0.35	2.528	3.923	0.506	2.59	4.197	0.483	2.5281	3.9225	0.5057
−0.3	2.267	3.902	0.407	2.035	3.432	0.396	2.2665	3.9017	0.4074
−0.25	1.246	2.308	0.275	1.535	2.731	0.317	1.2461	2.3076	0.2745
−0.2	1.257	2.334	0.275	1.306	2.409	0.281	1.2569	2.3344	0.2745
−0.15	1.255	2.313	0.276	1.222	2.292	0.268	1.2549	2.3127	0.2757
−0.1	1.279	2.377	0.277	1.195	2.253	0.263	1.2792	2.3766	0.2767
−0.05	1.243	2.313	0.272	1.185	2.24	0.261	1.2426	2.3132	0.2717
0.05	4.084	3.946	7.379	4.098	3.937	7.372	4.0836	3.9455	7.3786
0.1	2.847	2.549	5.836	2.581	2.735	5.957	2.8466	2.5487	5.8362
0.15	2.365	3.405	5.931	2.564	2.839	6.108	2.3648	3.405	5.9314
0.2	2.974	4.075	8.433	3.181	4.366	8.16	2.9743	4.0754	8.4334
0.25	4.95	6.124	12.82	4.392	7.355	12.18	4.9502	6.1243	12.8249
0.3	4.9	8.827	13.18	4.693	8.099	13.18	4.9004	8.8272	13.183
0.35	4.19	8.867	13.26	4.721	8.169	13.27	4.1904	8.8674	13.2624
0.4	4.545	7.918	12.63	4.725	8.201	13.29	4.5451	7.9175	12.6266
0.45	4.856	8.483	13.46	4.738	8.505	13.43	4.856	8.4831	13.4606
0.5	4.823	10.86	14.54	4.84	10.85	14.54	4.8225	10.8598	14.5434
$\sum Abs(error)$				3.1830	5.6320	2.0780	0.0055	0.0061	0.0224

process information by its dynamic state response”. A neural network consists of a number of very simple and highly interconnected processors called neurons, which are the analogs of the neurons in the brain (Zhang et al., 1995). The neurons are connected by a large number of weighted links, over which signals can pass. In the present application, three layers neural network (having an input layer, a hidden layer and an output layer) have been used, together with a tansigmoidal activation function and supervised training via a back-propagation technique, as shown in Fig. 2.

The input to ANN is the change of load ΔP_L, and the output is updated three control parameters K_P, K_I and K_D. The well known enhancement of introducing a momentum term in the weight updating formula has also been successfully applied to reduce training times and to help in avoiding premature convergence.

The weights of neural network are adapted depending on the error signal coming from the error between desired and actual three control parameters. To optimize the network, its error function is formulated in such a way that is quadratic in terms of the parameters to be estimated.

The error function E is defined as:

$$E = \frac{1}{2}\sum (K_r(m) - K_L(m))^2 \tag{2}$$

where K_r are the actual three controllers calculated from GA and K_L are the desired target at any time k, during each time interval from $m-1$ to m, the back-propagation algorithm is used to update the connective weights w according to the relation

$$w_{ij}(n+1) = w_{ij}(n) - \lambda \frac{\partial E}{\partial w_{ij}(n)} + \mu \Delta w_{ij}(n-1) \tag{3}$$

where λ is the learning rate, μ is the momentum factor, and n indicates the number of training iterations. A three-layer (input, hidden, and output) network is used for the neural controller.

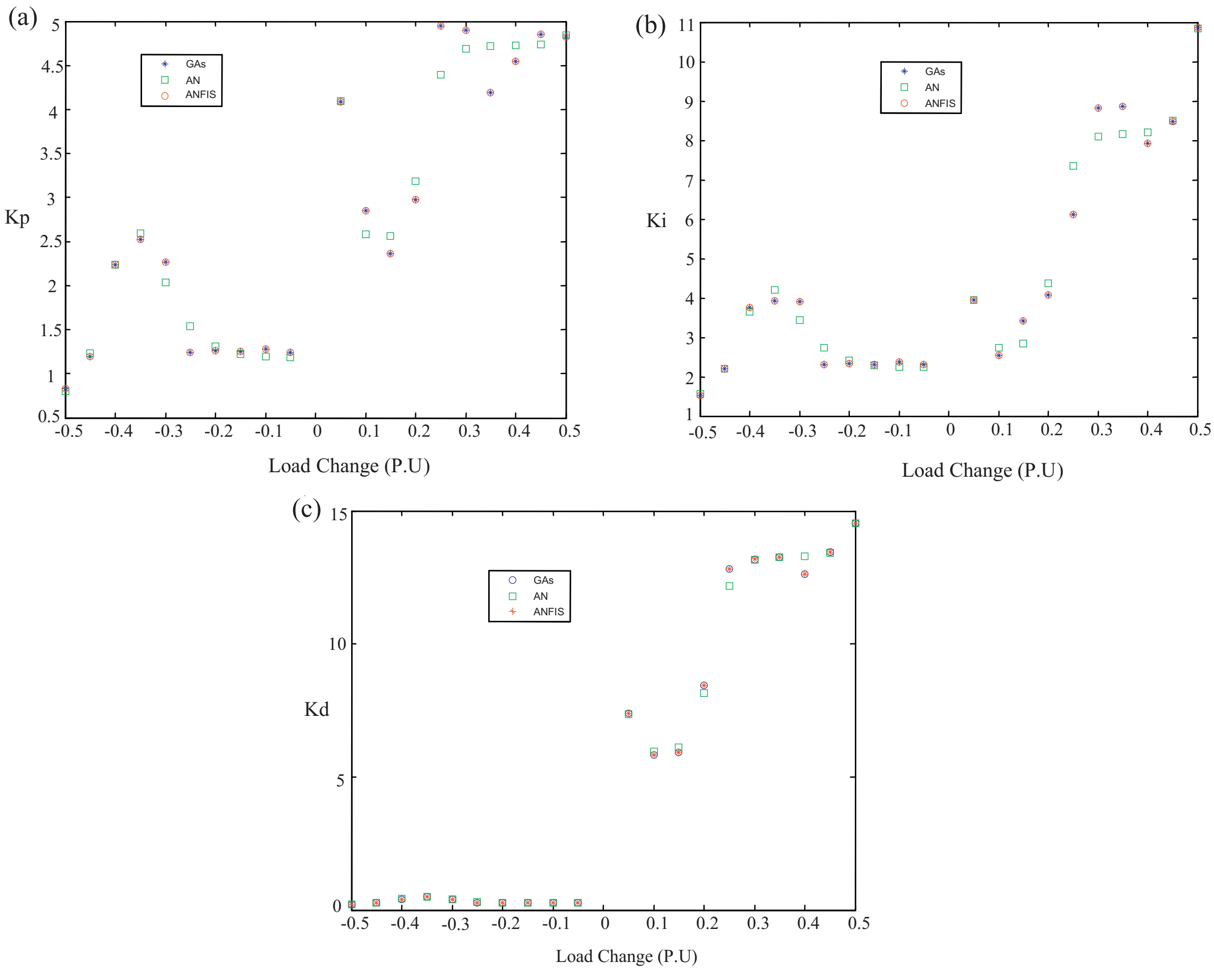

Fig. 4. (a) Proportional controllers from GAs, ANN and ANFIS. (b) Integral controllers from GAs, ANN and ANFIS. (c) Derivative controllers from GAs, ANN and ANFIS.

4.2. Adaptive Neuro-Fuzzy Inference Systems (ANFIS)

An Adaptive Neuro-Fuzzy Inference System (ANFIS) refers, in general, to an adaptive network which performs the function of a fuzzy inference system. The most commonly used fuzzy system in ANFIS architectures is the Sugeno model since it is less computationally exhaustive and more transparent than other models. A consequent membership function (MF) of the Sugeno model could be any arbitrary parameterized function of the crisp inputs, most likely a polynomial. Zero and first order polynomials are used as consequent MF in constant and linear Sugeno models, respectively. In addition, the defuzzification process in Sugeno fuzzy models is a simple weighted average calculation. The fuzzy space is divided via grid partitioning according to the number of antecedent MF, and each fuzzy region is covered with a fuzzy rule. On the other hand, each fixed and adaptive node of the network performs one function or sub-function of the Sugeno model, as shown in Fig. 3, such that the overall performance of the network is functionally the same as that of the fuzzy model (Pothiya and Tantaswadi, 2006).

The adaptive network employs an optimization algorithm to modify the parameters of the fuzzy inference system. The adaptation process aims at obtaining a set of parameters at which an error measure between the actual performance of the fuzzy model and a targeted set of training data is minimized. Classical optimization techniques, such as back propagation, could be used as well as hybrid algorithms (Panda et al., 2009). The total number of ANFIS modifiable parameters is a crucial factor of the computational effort required before the adaptation process is completed. Therefore, antecedent Gaussian MF, which is defined through two parameters only, is more preferable than other forms of MF.

Table 2
System response parameters with GAs, ANN and ANFIS.

ΔP_L	Algorithm								
	GA			ANN			ANFIS		
	ISE	IAE	ITAE	ISE	IAE	ITAE	ISE	IAE	ITAE
−0.5	0.1267	0.3289	0.1595	0.1348	0.3313	0.1495	0.1267	0.3289	0.1595
−0.45	0.0625	0.2036	0.0760	0.0601	0.2059	0.0887	0.0625	0.2036	0.0760
−0.4	0.0200	0.1066	0.0381	0.0192	0.1098	0.0448	0.0200	0.1066	0.0381
−0.35	0.0117	0.0896	0.0395	0.0118	0.0836	0.0310	0.0117	0.0896	0.0394
−0.3	0.0110	0.0771	0.0279	0.0125	0.0875	0.0323	0.0110	0.0771	0.0277
−0.25	0.0179	0.1084	0.0400	0.0133	0.0918	0.0357	0.0179	0.1084	0.0400
−0.2	0.0113	0.0858	0.0316	0.0108	0.0833	0.0324	0.0114	0.0858	0.0316
−0.15	0.0064	0.0650	0.0256	0.0067	0.0656	0.0240	0.0064	0.0650	0.0255
−0.1	0.0028	0.0422	0.0161	0.0031	0.0446	0.0162	0.0028	0.0422	0.0161
−0.05	0.0007	0.0217	0.0084	0.0008	0.0225	0.0082	0.0007	0.0217	0.0084
0.05	0.0001	0.0224	0.0894	0.0001	0.0228	0.0928	0.0001	0.0223	0.0885
0.1	0.0006	0.0639	0.2349	0.0006	0.0670	0.2695	0.0006	0.0638	0.2331
0.15	0.0012	0.0945	0.3858	0.0014	0.1003	0.4132	0.0012	0.0945	0.3858
0.2	0.0016	0.1159	0.5855	0.0013	0.1006	0.4324	0.0015	0.1137	0.5488
0.25	0.0016	0.1418	1.1014	0.0010	0.0872	0.3971	0.0013	0.1110	0.6391
0.3	0.0022	0.1667	1.3530	0.0022	0.1615	1.2343	0.0019	0.1432	1.0021
0.35	0.0028	0.1882	1.5485	0.0021	0.1338	0.7535	0.0025	0.1709	1.2860
0.4	0.0024	0.1353	0.6192	0.0038	0.2214	1.8125	0.0024	0.1353	0.6190
0.45	0.0034	0.1854	1.2975	0.0048	0.2482	2.0589	0.0034	0.1858	1.3003
0.5	0.0064	0.2975	2.6284	0.0065	0.2987	2.6460	0.0056	0.2689	2.2130

ANFIS combines the advantages of fuzzy systems and adaptive networks in one hybrid intelligent paradigm. The flexibility and subjectivity of fuzzy inference systems, when added to the optimization strength of adaptive networks, give ANFIS its remarkable power of modeling, learning, nonlinear mapping, and pattern recognition (Afzalian and D. A, 2000; Juang and Lu, 2005; Salem and Awadallah, 2014).

5. Results and discussions

In order to verify the effectiveness of the proposed adaptive control techniques, a wide range of operation is considered. Practically, the load changes between −0.1 p.u. and +0.1 p.u. However, a range of operation between −0.5 p.u. and +0.5 p.u. with a step of 0.05 p.u. (20 points) is considered in this study to ensure the robustness of the proposed adaptive control techniques.

The PID control parameters for each load are obtained by GAs in order to minimize ISE. These control parameters are used to train both ANN and ANFIS with these loading conditions in order to get on-line control parameters adaptation. The three control parameters obtained from GAs and those obtained from training both ANN and ANFIS are shown in Table 1. Moreover, the sum of absolute of errors between each control parameter and the corresponding ones obtained from both ANN and ANFIS at each loading condition are also depicted.

In the present work, three layers neural network (having an input layer with one input representing the load change, a hidden layer including 20 neurons and an output layer for the three control parameters) have been used, together with a tansigmoidal activation function and supervised training via a back-propagation technique. The mean square error (MSE) of the training data MSE stagnated at 0.258 after 7 training epochs. It should be mentioned that all Neural Networks are developed using NFTOOL Toolbox of MATLAB 7.8.

A single-input three-output ANFIS is designed with Gaussian antecedent membership function (MF), and first-order polynomial consequent MF. With three MF per input, the mean square error (MSE) of the training data did not stagnate even after 1000 training epochs. However, with five and seven MF per input, MSE stagnated at 0.000129 and 0.000103 after 64 and 16 training epochs, respectively. It should be mentioned that all ANFIS of the present work are developed using the Fuzzy Logic Toolbox of MATLAB 7.8. It is worth noting that three ANFIS models are designed for the three control parameters K_P, K_I, and K_D due to a limitation of ANFIS toolbox to have only one output per ANFIS.

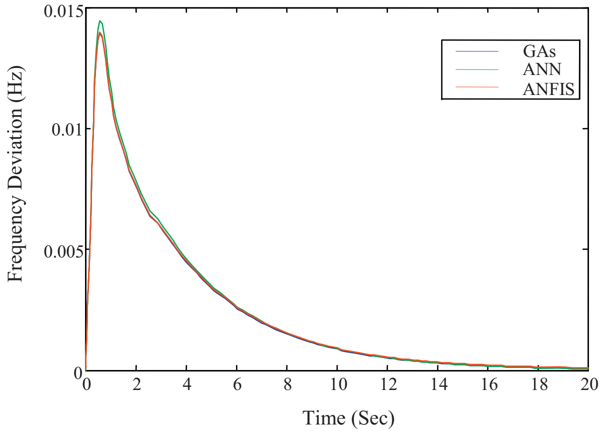

Fig. 5. Frequency deviation of the system at ΔP_L -5%.

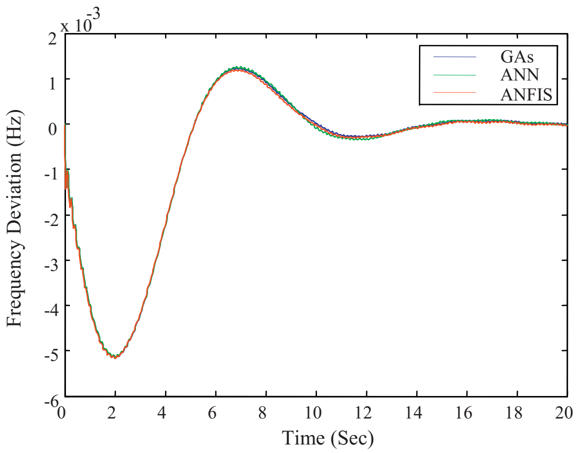

Fig. 6. Frequency deviation of the system at ΔP_L 5%.

From these results, it can be revealed that ANFIS gives better performance than ANN. Sum of absolute of errors for the three control parameters of ANFIS is less than those of ANN, where the absolute sum of errors for K_P obtained using ANFIS is 0.0055 while it is 3.1830 using ANN. Similarly, absolute sum of errors for K_I and K_D are 0.0061 and 0.0224, respectively, using ANFIS while they are 5.6320 and 2.0780, respectively, using ANN.

The three control parameters obtained from GAs and those obtained from training ANN and ANFIS are illustrated in Fig. 4. In this figure, it is clear that ANFIS gives accurate results comparing with ANN which has some deviations from the references values.

The performance measures (ISE, IAE and ITAE) for these loading cases with GAs, ANN and ANFIS are indicated in Table 2.

From Table 2, it is clear that the system performance using ANFIS, rather than ANN, is nearly identical to GAs. The frequency deviation of the system with control parameters obtained from GAs, ANN and ANFIS at selected operating points $\Delta P_L \pm 0.05$ and ± 0.15 p.u are depicted in Figs. 5–8.

To test the effectiveness of these trained two systems another two intermediate points, rather than 20 trained points are selected, at ± 0.125 p.u. The control parameters form ANN, ANFIS and GAs are shown in Table 3.

The system response with these two loading conditions (ISE, IAE and ITAE) with GAs, ANN and ANFIS are indicated in Table 4.

The frequency deviation of the system under these load changes is depicted in Figs. 9 and 10.

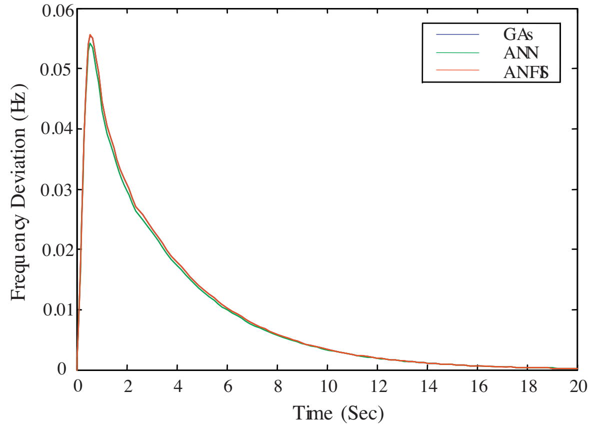

Fig. 7. Frequency deviation of the system at ΔP_L −20%.

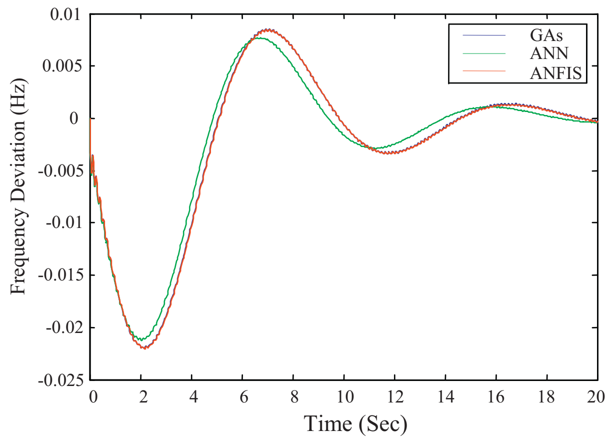

Fig. 8. Frequency deviation of the system at ΔP_L 20%.

Table 3
Control parameters from GAs, ANN and ANFIS.

ΔP_L	Algorithm								
	GA			ANN			ANFIS		
	K_P	K_I	K_D	K_P	K_I	K_D	K_P	K_I	K_D
−0.125	1.2705	2.3710	0.2766	1.1945	2.3411	0.2030	1.266	2.191	0.2787
0.125	4.8942	6.4779	10.9896	2.483	2.424	5.881	2.9545	3.0786	6.4876

Table 4
System response parameters with GAs, ANN and ANFIS.

ΔP_L	Algorithm								
	GA			ANN			ANFIS		
	ISE	IAE	ITAE	ISE	IAE	ITAE	ISE	IAE	ITAE
−0.125	0.0044	0.0528	0.0189	0.0057	0.0688	0.0343	0.0045	0.0572	0.0251
0.125	0.0003	0.0629	0.4601	0.0011	0.0908	0.3621	0.0008	0.0727	0.2735

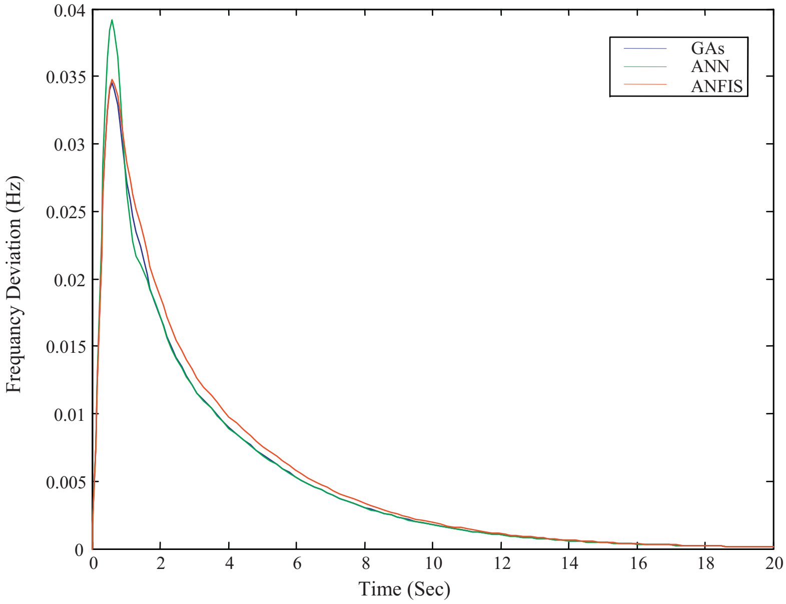

Fig. 9. Frequency deviation of the system at ΔP_L -12.5%.

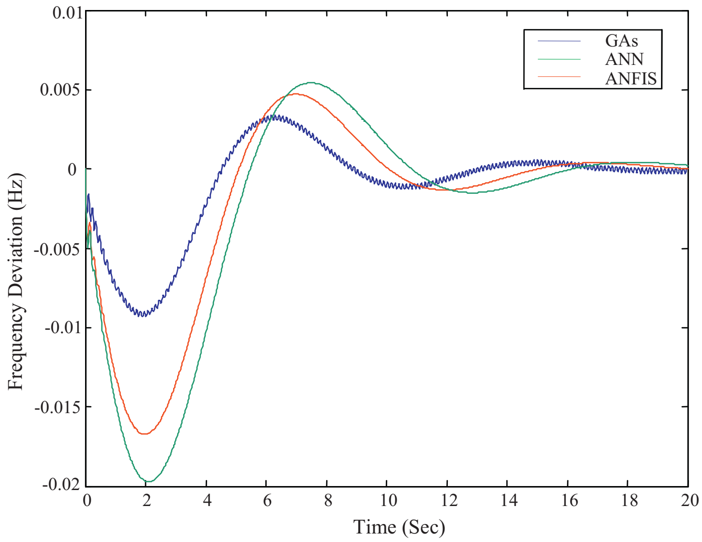

Fig. 10. Frequency deviation of the system at ΔP_L 12.5%.

6. Conclusion

This paper presents a design methodology based on ANN and ANFIS for an adaptive PID Load Frequency Control. A wide-range of load change is considered at which GA is employed to obtain the parameters of the PID controller yielding optimal responses. The data obtained through GA are used to train both ANN and ANFIS agent, which give the optimal controller parameters at any load point within the specified range. Both ANN and ANFIS testing denote notable effectiveness in learning and mapping the system characteristics. Testing both of the developed techniques show that the adaptive PID Load Frequency Control based ANN and ANFIS could preserve optimal performance over the whole loading range with superiority to ANFIS in terms of several performance measures such as Integral of Absolute of Errors (IAE), Integral of Time multiplied Absolute of Errors (ITAE), and Integral of Square of Errors (ISE).

7. Future work

Authors are going to check the proposed tuning methodology for much more complex system such as two area power system with weak tie line. Moreover, authors may apply other techniques instead of tuning a PID controller.

References

Afzalian, A., Linkens, D.A., 2000. Training of neuro fuzzy power system stabilizers using genetic algorithms. Int. J. Electr. Power Energy Syst. 22 (2), 93–102.

Aldden, M., Trinh, H., 1994. Load-frequency control of interconnected power systems via constrained feedback control schemes. Comp. Electr. Eng. 20 (1), 71–88.

Cox, E., 1992. Fuzzy fundamentals. IEEE Spectr. 29 (10), 58–61.

Elgerd, O.I., Fosha, C.E., 1970. Optimal megawatt-frequency control of multi-area electrical energy systems. IEEE Trans. PAS PAS 89 (April (4)), 556–563.

Haupt, R., Haupt, S.E., 1998. Practical Genetic Algorithms. John Wiley and Sons, San Francisco, CA, USA.

Hsu, Y., Cheng, C., 1991. Load frequency control using fuzzy logic. In: Int. Conf. on High Technology in the Power Industry, pp. 32–38.

Jang, J.-S., 1993. ANFIS: adaptive-network-based fuzzy inference system. IEEE Trans. Syst. Man Cybern. 23 (3), 665–685.

Jang, J.-S., Sun, C.-T., Mizutani, E., 1997. Neuro-Fuzzy and Soft Computing – A Computational Approach to Learning and Machine Intelligence. Prentice-Hall.

Juang, C.-F., Lu, C.-F., 2002. Power System Load Frequency Control With Fuzzy Gain Scheduling Designed by New Genetic Algorithms. IEEE, 0-7803-7280-8/02/$10.00.

Juang, C.-F., Lu, C.-F., 2005. Power system load frequency control by genetic fuzzy gain scheduling controller. J. Chin. Inst. Eng. 28 (6), 1013–1018.

Kundur, P., 1994. Power System Stability and Control. McGraw-Hill, New York.

Nanda, J., Kaul, B.L., 1978. Automatic generation control of an interconnected power system. IEE Proc. 125 (May (5)), 385–390.

Panda, G., Panda, S., Ardil, C., 2009. Hybrid neuro fuzzy approach for automatic generation control of two – area interconnected power system. Int. J. Comp. Intell. 5 (1), 80–84.

Pothiya, S., Tantaswadi, P., 2006. Design of optimal fuzzy logic based pi controller using multiple tabu search algorithm for load frequency control. Int. J. Control Autom. Syst. 4 (2), 155–164.

Saadat, H., 2005. Power System Analysis. Mcgraw-Hill International.

Sabahi, K., Nekoui, M.A., Teshnehlab, M., Aliyari, M., Mansouri, M., 2007. Load frequency control in interconnected power system using modified dynamic neural networks. In: 2007 Mediterranean Conference on Control and Automation, Athens, Greece, July 27–29.

Salem, F., Awadallah, M., 2014. Parameters estimation of photovoltaic modules: comparison of ANN and ANFIS. Int. J. Ind. Electron. Drives 1 (2), 121–129.

Talaq, J., Al-Basri, F., 1999. Adaptive fuzzy gain scheduling for load frequency control. IEEE Trans. Power Syst. 14 (February (1)).

Zhang, Y., Sen, P., Hearn, E.G., 1995. An on-line trained adaptive neural controller. IEEE Control Syst. Mag. (October), 67–75.